Name:Chao Jiang

ISE 5643 — Spring 2015 **HW 2** Due: Feb. 11

Problem 1

Use Golden Section Search to determine (within an interval of 0.8) the optimal solution to

$$\begin{aligned} \min \quad & x^2 + 2x \\ s.t. \quad & -3 \leq x \leq 3 \end{aligned}$$

Check your results by writing a code in MATLAB.

$$\begin{aligned}
x_1 &= 3 - 0.618 * (3 - (-3)) = -0.708 \\
x_2 &= -3 + 0.618 * (3 - (-3)) = 0.708 \\
f(x_1) &< f(x_2); f(x_1) = -0.9147, f(x_2) = 1.9173, \\
I_1 &= (-3, 0.708) \\
L_1 &= 0.708 - (-3) = 3.708 \\
x_3 &= 0.708 - 0.618 * 3.708 = -1.5835 \\
x_4 &= -0.708 \\
f(x_3) &> f(x_4); f(x_3) = -0.6595, f(x_4) = -0.9147 \\
I_2 &= (-1.5835, 0.708) \\
L_2 &= 0.708 - (-1.5835) = 2.2915 \\
x_5 &= -0.708 \\
x_6 &= -0.1674 \\
f(x_5) &< f(x_6); f(x_5) = -0.9147, f(x_6) = -0.3068 \\
I_3 &= (-1.5838, -0.1674) \\
L_3 &= -0.1674 - (-1.5838) = 1.4164 \\
x_7 &= -0.1674 - 0.618 * 1.416 = -1.0425 \\
x_8 &= -0.708 \\
f(x_7) &< f(x_8); f(x_7) = -0.9982, f(x_8) = -0.9147 \\
I_4 &= (-1.5838, -0.708) \\
L_4 &= (-0.708 - (-1.5838)) = 0.8758 \\
x_9 &= -0.708 - 0.618 * 0.8758 = -1.2492 \\
x_10 &= -1.0425 \\
f(x_9) &> f(x_10); f(x_9) = -0.9379, f(x_10) = -0.9982 \\
I_5 &= (-1.2492, -0.708) \\
L_5 &= (-0.708 - (-1.2492)) = 0.5412 < 0.8 \\
x^* &\in (-1.2492, -0.708)
\end{aligned}$$

- Using Matlab:

```
interation: 0
interval:(-3,3),x1=-0.708204,x2=0.708204,f(x1)=-0.914855,f(x2)=1.91796,length=6
interation: 1
interval:(-3,0.708204),x1=-1.58359,x2=-0.708204,f(x1)=-0.65942,f(x2)=-0.914855,length=3.7082
interation: 2
interval:(-1.58359,0.708204),x1=-0.708204,x2=-0.167184,f(x1)=-0.914855,f(x2)=-0.306418,length=2.2918
interation: 3
interval:(-1.58359,-0.167184),x1=-1.04257,x2=-0.708204,f(x1)=-0.998188,f(x2)=-0.914855,length=1.41641
interation: 4
interval:(-1.58359,-0.708204),x1=-1.24922,x2=-1.04257,f(x1)=-0.937888,f(x2)=-0.998188,length=0.875388
interation: 5
interval:(-1.24922,-0.708204),x1=-1.04257,x2=-0.914855,f(x1)=-0.998188,f(x2)=-0.99275,length=0.54102
```

Problem 2

Consider the minimization of the function

$$f(x) = (10x^3 + 3x^2 + x + 5)^2$$

Use Swanns search to bound the minimum starting at

(a) $x_0 = 0, \Delta = 1$

$$\begin{aligned} f(x_0 + \delta) &= 361 \\ f(x_0) &= 25 \\ f(x_0 - \delta) &= 9 \end{aligned}$$

So we will continue our search on the left of the axial.

$$\begin{aligned} x_k = x_{k-1} + 2^k \Delta \\ x_1 = -1 - 2 = -3; \\ f(-3) = 58081 \end{aligned}$$

So we get our optimal minimum at interval $(-3, 0)$

(b) $x_0 = 0, \Delta = 5$

$$\begin{aligned} f(x_0+\delta) &= 1782225 \\ f(x_0-\delta) &= 1380625 \end{aligned}$$

Compare the two bounds and comment From (a) we get the bound $x^* \in (-3, 0)$ which is better than (b) whose bound is $x^* \in (-5, 5)$

Problem 3

Use the best bound obtained in problem 2 to carry out a single variable search to minimize $f(x)$ by (a) golden section and (b) interval halving. Each search method is to use 5 functional evaluations only. Compare the final search intervals obtained by the two methods.

we minimize function $f(x) = (10x^3 + 3x^2 + x + 5)^2$, with $x^* \in (-3, 0)$ by golden section and interval halving within 5 functional evaluations.

(a) Golden section

```
interation: 0
interval:(-3,0),x1=-1.8541,x2=-1.1459,f(x1)=2528.02,f(x2)
    =52.6094,length=3
interation: 1
interval:(-1.8541,0),x1=-1.1459,x2=-0.708204,f(x1)
    =52.6094,f(x2)=5.0375,length=1.8541
interation: 2
interval:(-1.1459,0),x1=-0.708204,x2=-0.437694,f(x1)
    =5.0375,f(x2)=18.4772,length=1.1459
interation: 3
interval:(-1.1459,-0.437694),x1=-0.875388,x2=-0.708204,f(
    x1)=0.0810051,f(x2)=5.0375,length=0.708204
interation: 4
interval:(-1.1459,-0.708204),x1=-0.978714,x2=-0.875388,f(
    x1)=6.15031,f(x2)=0.0810051,length=0.437694
interation: 5
interval:(-0.978714,-0.708204),x1=-0.875388,x2=-0.811529,f
    (x1)=0.0810051,f(x2)=0.671808,length=0.27051
```

(b) Interval Halving

```
interation: 0,RF=0
interval:(-3,0),x1=-2.25,xm=-1.5,x2=-0.75,f(x1)=9210,f(xm)
    =552.25,f(x2)=2.9541,length=3
interation: 1,RF=2
```

```
interval:(-1.5,0),x1=-1.125,xm=-0.75,x2=-0.375,f(x1)=43.1177,f(xm)=2.9541,f(x2)=20.4262,length=1.5
interation: 2,RF=4
interval:(-1.125,-0.375),x1=-0.9375,xm=-0.75,x2=-0.5625,f(x1)=2.37322,f(xm)=2.9541,f(x2)=13.01,length=0.75
interation: 3, RF=5
interval:(-1.125,-0.75),x1=-1.03125,xm=-0.9375,x2=-0.84375,f(x1)=14.5003,f(xm)=2.37322,f(x2)=0.0813489,length=0.375
interation: 4,RF=7
interval:(-0.9375,-0.75),x1=-0.890625,xm=-0.84375,x2=-0.796875,f(x1)=0.331245,f(xm)=0.0813489,f(x2)=1.09814,length=0.1875
```

The fractional reduction of golden research is $0.27051/3 = 0.0902$; the interval halving' s RF is $0.375/3 = 0.125$.

Problem 4

(a) Using Golden Section Search and 10 functional evaluations, what is the interval reduction we can achieve?

```
interation: 0
interval:(-3,0),x1=-1.8541,x2=-1.1459,f(x1)=2528.02,f(x2)=52.6094,length=3
interation: 1
interval:(-1.8541,0),x1=-1.1459,x2=-0.708204,f(x1)=52.6094,f(x2)=5.0375,length=1.8541
interation: 2
interval:(-1.1459,0),x1=-0.708204,x2=-0.437694,f(x1)=5.0375,f(x2)=18.4772,length=1.1459
interation: 3
interval:(-1.1459,-0.437694),x1=-0.875388,x2=-0.708204,f(x1)=0.0810051,f(x2)=5.0375,length=0.708204
interation: 4
interval:(-1.1459,-0.708204),x1=-0.978714,x2=-0.875388,f(x1)=6.15031,f(x2)=0.0810051,length=0.437694
interation: 5
interval:(-0.978714,-0.708204),x1=-0.875388,x2=-0.811529,f(x1)=0.0810051,f(x2)=0.671808,length=0.27051
interation: 6
interval:(-0.978714,-0.811529),x1=-0.914855,x2=-0.875388,f(x1)=1.1256,f(x2)=0.0810051,length=0.167184
```

```
interation: 7
interval:(-0.914855,-0.811529),x1=-0.875388,x2=-0.850996,f(x1)=0.0810051,f(x2)=0.025191,length=0.103326
interation: 8
interval:(-0.875388,-0.811529),x1=-0.850996,x2=-0.835921,f(x1)=0.025191,f(x2)=0.175771,length=0.0638587
interation: 9
interval:(-0.875388,-0.835921),x1=-0.860313,x2=-0.850996,f(x1)=5.48891e-05,f(x2)=0.025191,length=0.0394669
interation: 10
interval:(-0.875388,-0.850996),x1=-0.866071,x2=-0.860313,f(x1)=0.0125567,f(x2)=5.48891e-05,length=0.0243919
```

After 10 functional evaluations, our interval length becomes to 0.0243919.

(b) For achieving the same interval reduction, how many function evaluations would be needed in the interval-halving method?

```
interation: 0,RF=0
interval:(-3,0),x1=-2.25,xm=-1.5,x2=-0.75,f(x1)=9210,f(xm)=552.25,f(x2)=2.9541,length=3
interation: 1,RF=2
interval:(-1.5,0),x1=-1.125,xm=-0.75,x2=-0.375,f(x1)=43.1177,f(xm)=2.9541,f(x2)=20.4262,length=1.5
interation: 2,RF=4
interval:(-1.125,-0.375),x1=-0.9375,xm=-0.75,x2=-0.5625,f(x1)=2.37322,f(xm)=2.9541,f(x2)=13.01,length=0.75
interation: 3, RF=5
interval:(-1.125,-0.75),x1=-1.03125,xm=-0.9375,x2=-0.84375,f(x1)=14.5003,f(xm)=2.37322,f(x2)=0.0813489,length=0.375
interation: 4,RF=7
interval:(-0.9375,-0.75),x1=-0.890625,xm=-0.84375,x2=-0.796875,f(x1)=0.331245,f(xm)=0.0813489,f(x2)=1.09814,length=0.1875
interation: 5,RF=9
interval:(-0.890625,-0.796875),x1=-0.867188,xm=-0.84375,x2=-0.820312,f(x1)=0.017561,f(xm)=0.0813489,f(x2)=0.46028,length=0.09375
interation: 6, RF=10
interval:(-0.890625,-0.84375),x1=-0.878906,xm=-0.867188,x2=-0.855469,f(x1)=0.123074,f(xm)=0.017561,f(x2)=0.00631424,length=0.046875
interation: 7,RF=12
```

```
interval:(-0.867188,-0.84375),x1=-0.861328,xm=-0.855469,x2
    =-0.849609,f(x1)=0.000662737,f(xm)=0.00631424,f(x2)
    =0.0335299,length=0.0234375
interation: 8, RF=13
interval:(-0.867188,-0.855469),x1=-0.864258,xm=-0.861328,
    x2=-0.858398,f(x1)=0.00623057,f(xm)=0.000662737,f(x2)
    =0.000731947,length=0.0117188
interation: 9,RF=15
interval:(-0.864258,-0.858398),x1=-0.862793,xm=-0.861328,
    x2=-0.859863,f(x1)=0.00273421,f(xm)=0.000662737,f(x2)
    =4.96094e-07,length=0.00585938
```

For interval-halving method, 12 function evaluations are needed to achieving the same interval reduction with $RF = 0.0081$. which is close to the theory equation from textbook that:

$$FR(N) = \begin{cases} (0.5)^{\frac{N}{2}} & \text{for Interval-halving method} \\ (0.618)^{N-1} & \text{for Golden section method} \end{cases}$$

Matlab Code

All 3 files can be download at my GitHub[1].
Function:

```
function y=f(x)
%y = (10*x^3+3*x^2+x+5)^2;
y = x^2+2*x;
end
```

Golden Section Search:

```
% mailto:chao@ou.edu
% Golden Section Search for minimization of function f(x)
    within initial
% interval (a,b) at the accuracy value of epsilon.

% initial parameters
a = -3;                                        % start of
    interval
b = 0;                                         % end of
    interval
epsilon = 0.00001;                             % accuracy
    value
```

[1]https://github.com/chao92/Engineering-Optimization

```
ratio = double((sqrt(5)-1)/2);                    % golden proportion coefficient , aroud 0.618
k = 0;                                            % number of iterations
iter = 10;                                         % maximun number of iterations

len = b-a;
x1 = b - ratio * len;                             % computing x1 ;
x2 = a + ratio * len;                             % computing x2 ;

f1=f(x1);
f2=f(x2);
RF=0;
fprintf(1,'interation: %d,RF=%d',k,RF);
fprintf(1,'interval:(%g,%g),x1=%g,x2=%g,f(x1)=%g,f(x2)=%g,length=%g\n',a,b,x1,x2,f1,f2,len);
% search
while(((b-a) > epsilon) && (k<iter))

    % evaluate f(x1) and f(x2)
    if(f1<f2)                                     % if f(x1)<f(x2), then drop interval (x2,b);
            b = x2;                               % repalce b with x2;
            len = x2-a;                           % length becomes b-a=x2-a;
            x2 = x1;                              % replace the next iteration x2 with x1;
            x1 = b - ratio * len;                 % computing x1 using the equation of computing x1;
            f1=f(x1);
            f2=f(x2);
            RF=RF+1;
            k=k+1;
            fprintf(1,'interation: %d,RF=%d',k,RF);
            fprintf(1,'interval:(%g,%g),x1=%g,x2=%g,f(x1)=%g,f(x2)=%g,length=%g\n',a,b,x1,x2,f1,f2,len);
    else

            a = x1;
```

```
                len = b -a;
                x1 = x2;
                x2 = a + ratio * len;
                f1=f(x1);
                f2=f(x2);
                RF=RF+1;
                k=k+1;
                fprintf(1,'interation: %d,RF=%d',k,RF);
                fprintf(1,'interval:(%g,%g),x1=%g,x2=%g,f(x1)=%g,f(x2)=%g,length=%g\n',a,b,x1,x2,f1,f2,len);
    end

end
fprintf(1,'interation: %d,RF=%d',k,RF);
fprintf(1,'interval:(%g,%g),x1=%g,x2=%g,f(x1)=%g,f(x2)=%g,length=%g\n',a,b,x1,x2,f1,f2,len);
```

Interval-Halving Search:

```
% mailto:chao@ou.edu
% Interval Halving Search for minimization of function f(x) within initial
% interval (a,b) at the accuracy value of epsilon.

a = -3;                                     % start of interval
b = 0;                                      % end of interval
epsilon = 0.00001;                          % accuracy value
k = 0;                                      % number of iterations
iter = 10;                                  % maximun number of iterations

len = b-a;
xm = (a + b)/2;
x1 = a+len/4;                               % computing x1;
x2 = b-len/4;                               % computing x2;
RF=0;
fm=f(xm);
f1=f(x1);
```

```
f2=f(x2);
fprintf(1,'interation: %d,RF=%d\n',k,RF);
fprintf(1,'interval:(%g,%g),x1=%g,xm=%g,x2=%g,f(x1)=%g,f(xm)=%g,f(x2)=%g,length=%g\n',a,b,x1,xm,x2,f1,fm,f2,len);
while(((b-a) > epsilon) && (k<iter))
    %fprintf(1,'interation: %d\n',k);
    % fprintf(1,'interval:(%g,%g),x1=%g,xm=%g,x2=%g,f(x1)=%g,f(xm)=%g,f(x2)=%g,length=%g\n',a,b,x1,xm,x2,f1,fm,f2,len);
    % evaluate f(x1) and f(x2)
     if(f1<fm)                                  % if f(x1)<f(xm), then drop half of the interval (xm,b);
        b = xm;
        xm = x1;

        len = b-a;
        x1 = a+len/4;                           % computing x1;
        x2 = b-len/4;                           % computing x2;
        fm=f(xm);
        f1=f(x1);
        f2=f(x2);
        k=k+1;
        RF=RF+1;
        fprintf(1,'interation: %d, RF=%d\n',k,RF);
        fprintf(1,'interval:(%g,%g),x1=%g,xm=%g,x2=%g,f(x1)=%g,f(xm)=%g,f(x2)=%g,length=%g\n',a,b,x1,xm,x2,f1,fm,f2,len);
     else                                       % if f(x1)>f(xm), we need to continue compare f(xm) with f(x2)
        if(f2<fm)
            a = xm;
            xm = x2;
            len = b-a;
            x1 = a+len/4;                       % computing x1;
            x2 = b-len/4;                       % computing x2;
            fm=f(xm);
            f1=f(x1);
            f2=f(x2);
            k=k+1;
```

```
            RF=RF+2;
            fprintf(1,'interation: %d,RF=%d\n',k,RF);
            fprintf(1,'interval:(%g,%g),x1=%g,xm=%g,x2=%g,f(x1)=%g,f(xm)=%g,f(x2)=%g,length=%g\n',a,b,x1,xm,x2,f1,fm,f2,len);
        else                                    % this means that f(x2)>=f(xm) so our interval will be (x1,x2)
            a = x1;
            b = x2;
            len = b-a;
            xm = (a + b)/2;
            x1 = a+len/4;                       % computing x1;
            x2 = b-len/4;                       % computing x2;
            fm=f(xm);
            f1=f(x1);
            f2=f(x2);
            k=k+1;
            RF=RF+2;
            fprintf(1,'interation: %d,RF=%d\n',k,RF);
            fprintf(1,'interval:(%g,%g),x1=%g,xm=%g,x2=%g,f(x1)=%g,f(xm)=%g,f(x2)=%g,length=%g\n',a,b,x1,xm,x2,f1,fm,f2,len);
        end
    end
end
fprintf(1,'interation: %d,RF=%d\n',k,RF);
fprintf(1,'interval:(%g,%g),x1=%g,xm=%g,x2=%g,f(x1)=%g,f(xm)=%g,f(x2)=%g,length=%g\n',a,b,x1,xm,x2,f1,fm,f2,len);
```